

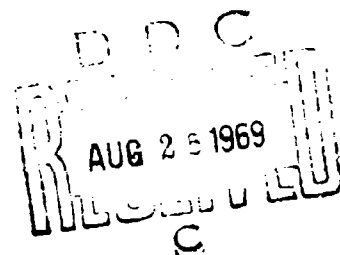
AD 691 834

THE NEW ROLE OF MANAGEMENT INFORMATION SYSTEMS

J. GOSDEN
E. RAICHELSON

APRIL 1969

THE MITRE CORPORATION



This document has been approved for public
release and sale; its distribution is unlimited.

DATA MANAGEMENT SERIES NO. 3

126

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The MITRE Corporation 1820 Dolley Madison Blvd. McLean, Virginia 22101		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP N/A	
3. REPORT TITLE The New Role of Management Information Systems			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) N/A			
5. AUTHOR S: (First name, middle initial, last name) John A. Gosden Eugene Raichelson			
6. REPORT DATE April 1969		7a. TOTAL NO. OF PAGES 26	7b. NO. OF REFS 7
8a. CONTRACT OR GRANT NO. F19628-68-C-0365		9a. ORIGINATOR'S REPORT NUMBER(S) MTP-332	
b. PROJECT NO. 33.4.B		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None	
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES N/A		12. SPONSORING MILITARY ACTIVITY Dep. Dir., NMCSTS, Defense Communications Agency, South Court House Road, Arlington, Virginia 22204, Attn: Mr. Reynold Thomas, Jr., Code 912	
13. ABSTRACT This paper is based upon a talk delivered at the UCLA/Informatics Symposium on "Interactive Computers for Controlling Machines and Influencing People: Setting the Specifications for the Fourth Generation," on March 27, 1969. It discusses the problems of "fourth generation" information systems and treats two basic problems: diversity of users and diversity of data. The paper concludes with a description of the characteristics needed in future information systems.			

~~UNCLASSIFIED~~
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Data management systems						
Management information systems						
User requirements						
System design						
Data interchange						
Data definition language						

UNCLASSIFIED
Security Classification

MTP-332

**THE NEW ROLE
OF MANAGEMENT
INFORMATION
SYSTEMS**

**J. GOSDEN
E. RAICHELSON**

APRIL 1969

THE MITRE CORPORATION

The work reported in this document was sponsored by
the Defense Communications Agency under Contract
F 19628-68-C-0365

This document has been approved for public
release and sale, its distribution is unlimited.

ABSTRACT

This paper is based upon a talk delivered at the UCLA/ Informatics Symposium on "Interactive Computers for Controlling Machines and Influencing People: Setting the Specifications for the Fourth Generation," on March 27, 1969. It discusses the problems of "fourth generation" information systems and treats two basic problems: diversity of users and diversity of data. The paper concludes with a description of the characteristics needed in future information systems.

INTRODUCTION

We define a management information system as a data management system especially devoted to the handling of data for management. This paper addresses the requirements for management information systems in the "fourth generation."

THE BASIC PROBLEM

Management information systems are probably the most interesting and most difficult applications of data management systems, primarily because of the ill-defined nature of management itself. A manager's job is, first, to develop plans to carry out company policies which supervisors and foremen in turn execute. Then, whenever a situation arises that is not covered by his plan, or when the work does not seem to be going according to plan, it is the manager's job to make corrections. When a particular exceptional situation begins to recur, it becomes "standardized" and, by that time, a manager should have developed a standardized correction or answering procedure. When such a procedure can be initiated automatically by a supervisor or foreman, it is no longer a management problem. Because a manager deals with exceptional cases, it is difficult to predict and define clearly what he is going to do, what data he is going to need, and how he is going to use it. Hence, we come again to the famous phrase, "management by exception." This is what makes implementing a management information system so difficult. The analyst does not know what the exceptions are going to be until they occur. Moreover, until an exception occurs, the kind of data that will be needed to handle it is also unknown. The analyst does not know what kinds of alternatives will be developed, what can be explored and examined, nor what kind of historical data will be relevant until he knows which

manager will handle the problem and how he "operates."

Therefore, a management information system must be general in purpose, highly flexible, and capable of adapting to new needs. In particular it must be able to deal with three kinds of variety: (1) many diverse kinds of users, (2) many diverse levels of users, and (3) many diverse kinds of data (see Figure 1).

Diverse Users

We can expect future management information systems to deal with a wide variety of users. Present data management system state-of-the-art has developed to the point where major technical problems can have several feasible solutions; and the economics are now such that with on-line storage and simple consoles a data management system can fit in well with all kinds of management situations (see Figure 2). We should expect a data management system to hold at least a coordinated set of files covering a single operating unit, even if it will not in the near future be an integrated data base.

Figure 3 shows many ways that users may vary. The first way users will vary will be in their disciplines. Within a production manufacturing industry, for example, we expect the same data base to be accessed by accountants, production engineers, design engineers, maintenance engineers, salesmen, economists, planners and division directors.

The second way users, particularly managers, will vary will be in their "style" of operation. The way one person manages can be very different from another. A management information system cannot be built on the hypothesis that the theory of management or that the "style of the manager" will be

● DIVERSE KINDS OF USER

ENGINEER, ECONOMIST, SALESMAN, MAINTENANCE, . . .

● DIVERSE USER FACILITIES

PRODUCTION, ENGINEERING DESIGN, MANAGEMENT,
PLANNERS, OPS RESEARCH, FINANCE

● DIVERSE DATA QUALITIES

DIFFERENT SOURCES (INVENTORY, TIME CARDS, ESTIMATES, . . .)
SEVERAL LEVELS OF AGGREGATION (BY TIME, BY GROUPING)
NON-STANDARD DATA ELEMENTS (EXTERNAL SOURCES, NEW
SYSTEMS)

SINGLE GENERAL INTEGRATED SOLUTION LEADS TO "KLUDGE"

FIGURE 1
MIS REQUIREMENTS

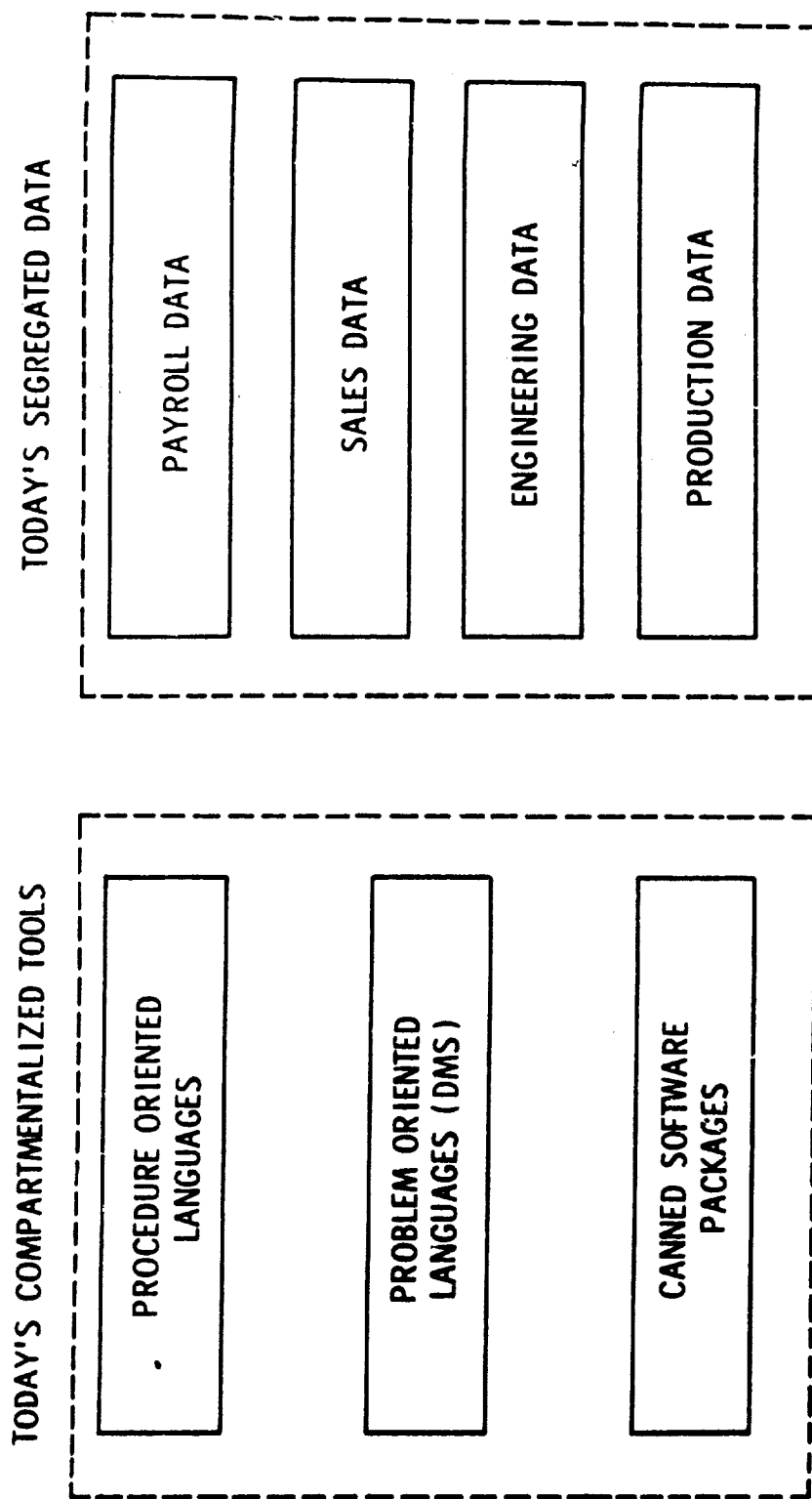


FIGURE 2
TODAY'S MANAGEMENT INFORMATION SYSTEM

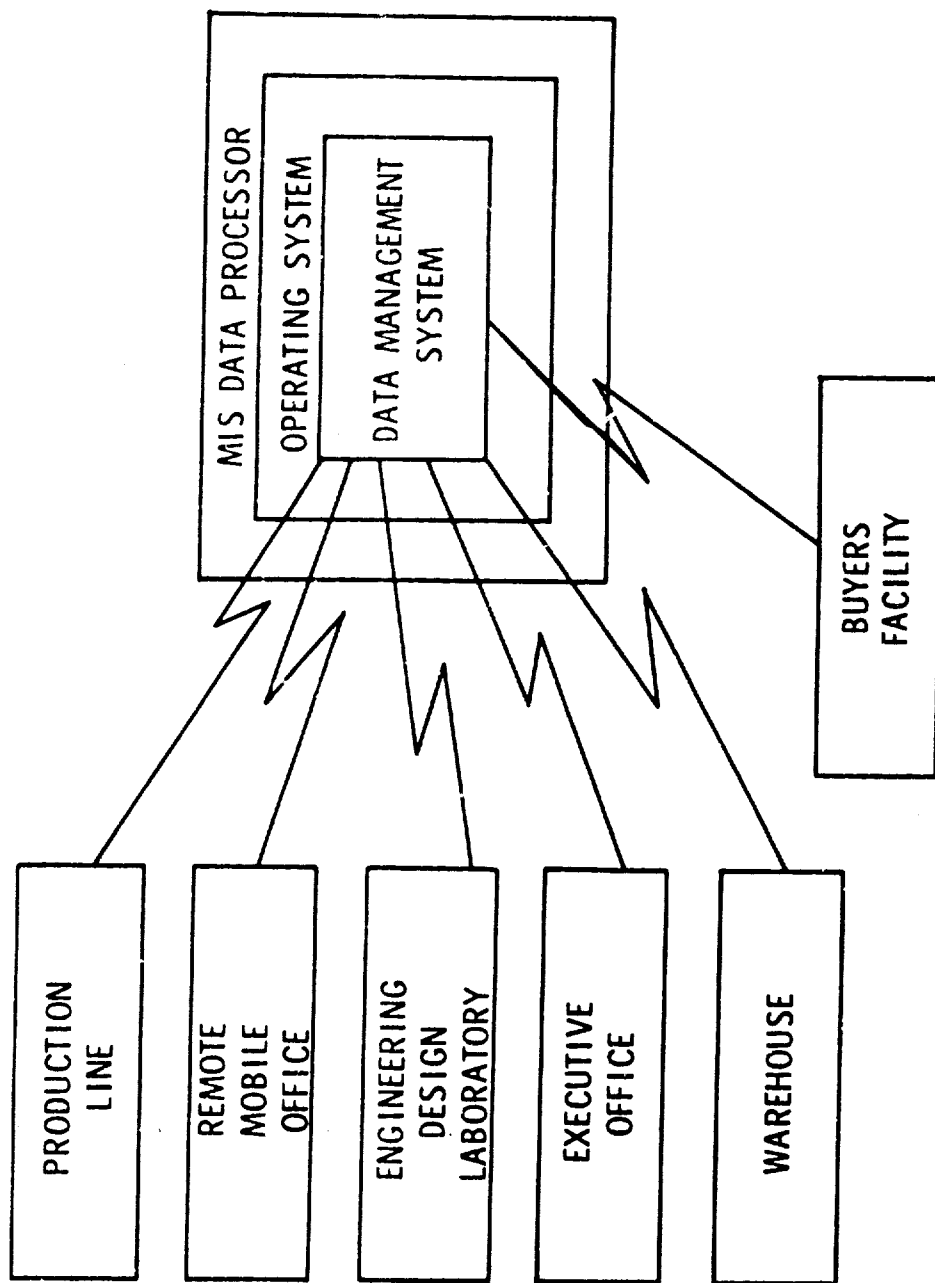


FIGURE 3
SOLUTION I-SEPARATE INTERFACE FOR EACH TYPE OF USER

uniform from one individual to another, even among those occupying the same nominal position, simultaneously and one after another. Furthermore, it must be able to deal with many levels of users. Some people will use the system frequently; others infrequently. Some will use it directly, others will employ "staff" to use it for them. We also have to deal with many levels of sophistication in the users. Some will want to exploit the tool. Already we know that we must be able to satisfy both types. It begins to look as though we need the most general-purpose system that can exist in the world. This immediately raises another difficulty. Those of you who have been involved with large-scale problems will know that the completely general-purpose solution often leads to our friend "kludge."

The straightforward solution to the problem of many different kinds of users will be the development of a variety of different user-oriented or problem-oriented languages. We are beginning to see these emerging now and can expect a continuing development of these kinds of languages as the present-day data management systems begin to be altered in response to customer demand. These languages will be developments from and will retain a lot of the structure of the "query" systems that we have today. However, we also expect various kinds of users to develop their own jargon, abbreviations, and other short-cuts in their use of these languages.

One of the easier developments to predict is the way that the different levels of user can be handled. Already in on-line interfaces to current data management systems we have seen the development of a variety of techniques to handle different levels of user. For example, we can identify

approximately three levels of sophistication of dialogue between man and a data management system (see Figure 4).

The first and simplest technique is the "tutorial" form in which the system leads the man "by the hand," listing the various choices open to him and describing their differences. This is a very relaxed and verbose kind of dialogue that essentially trains the user "on the job."

The second technique is also a "multichoice" approach. But in this case the options are presented to the man in an array with no dialogue; hence, he can quickly choose one option and move rapidly on. This may be done by function keys or lists of options on a display which he points to with a light gun.

The third technique allows the man to proceed ahead at his own pace without waiting for the computer to display the options for him. This is particularly useful with typewriter consoles where it is not possible to provide the options as rapidly as a display can.

Whatever the level of user involved, it will take time before all these levels are well developed for each area. Care has to be taken that a compatible set of levels exist, so that the "rules of the game" do not change as a user moves from one level to another, or even from one type of usage to another. An on-line system should not be sensitive to trivial errors. Default conditions must exist for missing cases. When obvious errors occur, they should be displayed immediately to the man so that he can correct them. Probably most important of all is a "must" item, a "help" facility: the last thing we want to have surrounding the console is a stack of manuals.

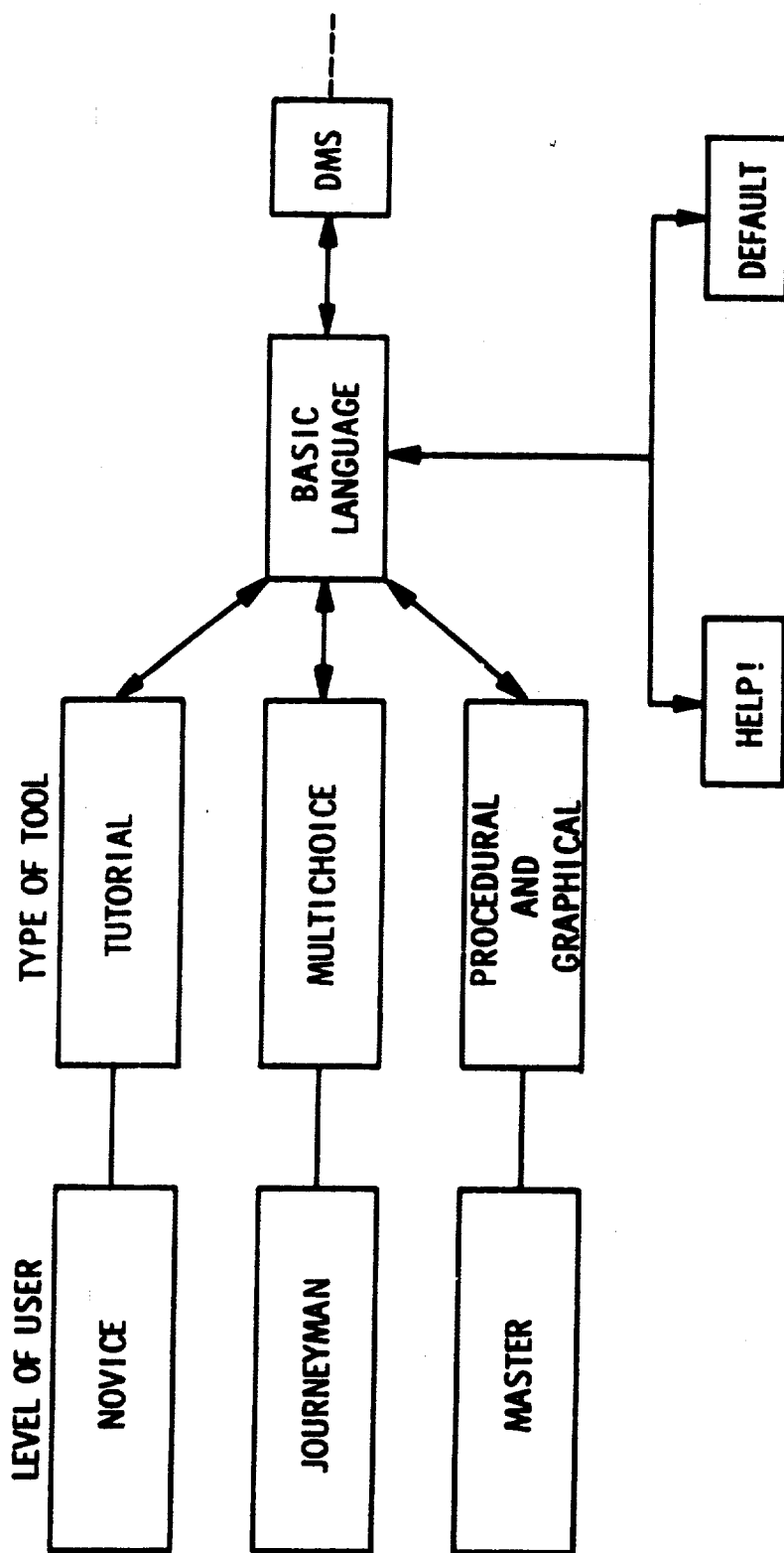


FIGURE 4
SOLUTION II SEVERAL TOOLS FOR 4TH GENERATION USERS

Diverse Data

The second major problem facing management information systems is the great variety of data to be handled. The phrases "integrated data base," "data pool," and "data bank," all arose from the realization that managers need data from many different sources in an organization--from the production lines, from the planning offices, from the finance centers, from the sales force, and from the distribution systems. In general, we do not know which data the manager will want in advance, nor which kinds may need to be combined or correlated. We therefore need a system which will be able to pick data from a variety of different places and put it together in ways that have not been anticipated.

Figure 5 shows that Data Management System software must be available and possess certain capabilities.

First the data must be available in order to be collected. In the case of a new and novel use, data will not have been collected for it. If the data is present, it will be because it was needed for some other explicit purpose, probably a mundane purpose such as payroll, cost accounting, billing or inventory control. The system must be able to pick pieces of data from these various areas and bring them together for an analysis by a manager or one of his staff. This might not be difficult if all the EDP work has been done in one uniform way, say using COBOL, on one kind of machine, within one corporation, and according to an agreed set of formats. That, however, is the exception. Again, what we are looking for is a solution of the difficult cases and Figures 6 and 7 indicate paths towards the solution. We need to be able to handle data bases that grew up under different programming languages for different users. We need to be able to handle data that originated in

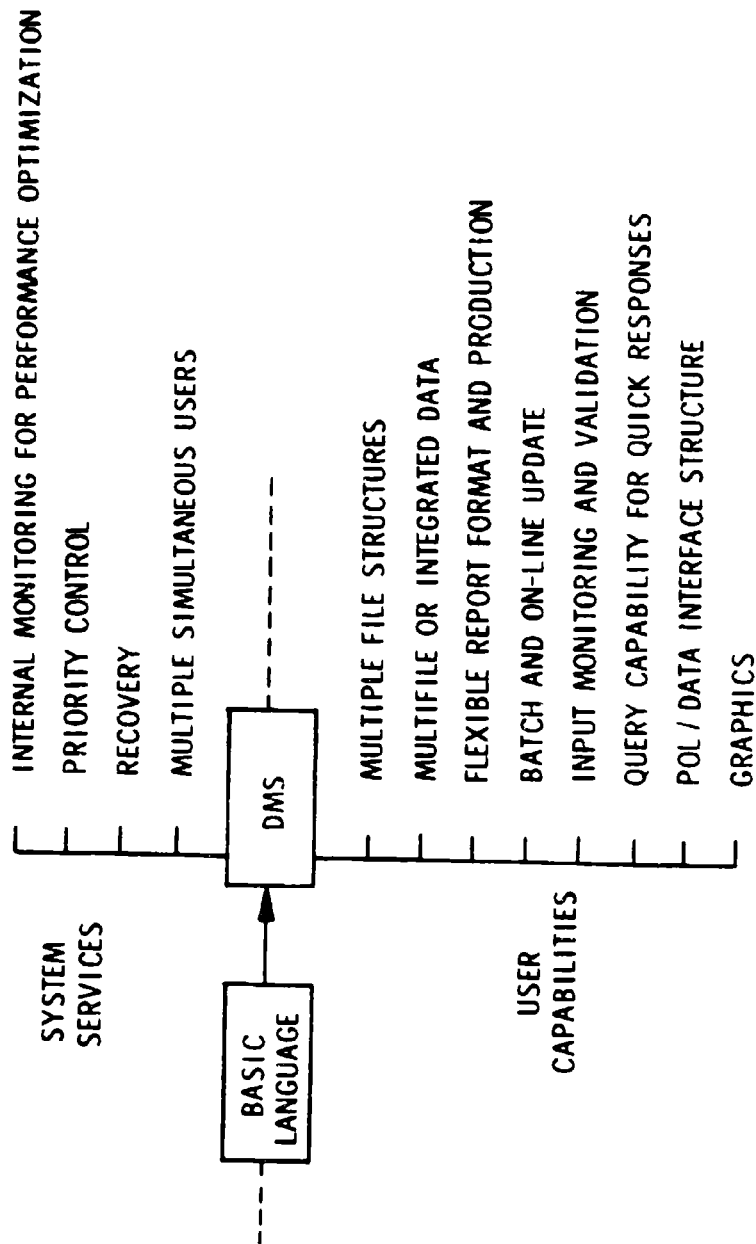


FIGURE 5
SOLUTION III - POWERFUL DMS PROCESSING CAPABILITIES

- MONOLITHIC: ● ALL DATA CREATED USING LIKE SOFTWARE,
PROCEDURES AND HARDWARE
- MULTILITHIC: ● DATA DESCRIPTION LANGUAGE FOR INTERCHANGE
OF DATA CREATED IN DIFFERENT ENVIRONMENTS:
- HARDWARE
 - LANGUAGE
 - FILE STRUCTURE
 - FORMAT

FIGURE 6
SOLUTION IV - SOLVE THE DATA INTERCHANGE PROBLEM

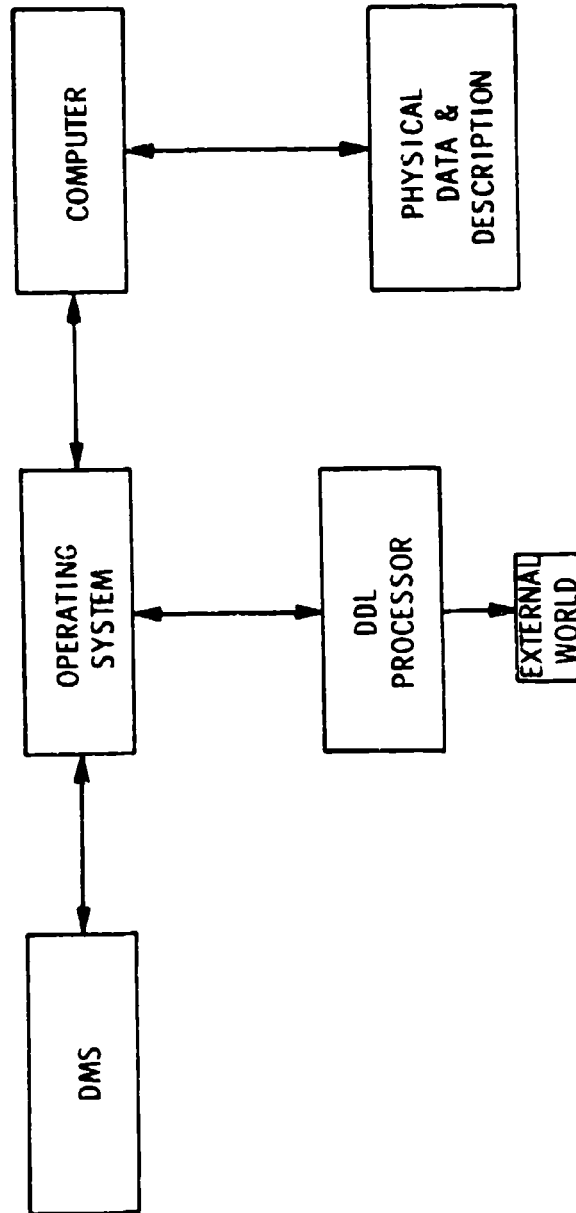


FIGURE 7
SOLUTION IV - DATA INTERCHANGE DIAGRAM

different systems, and was developed on different machines. Furthermore, in this age of decentralization of major divisions or groups of divisions within a corporation, and the emergence of conglomerates, we need to be able to bring together data from completely independent sources that were created when no thought had been given to the problem of an "integrated data base" for them. If we characterize the simple solution as a "monolithic" solution, where everyone uses the same machine, the same procedures, the same kinds of formats, then we can characterize the really difficult problem as the "multilithic" problem in which we must be prepared to bring together items from disparate systems. This is the way the world works and we have only just begun to realize the problem of compatibility in transferring data from one place to another. We call this the problem of data interchange.

Data interchange, in the context of this definition, occurs in the development of computer networks and data bases (e. g., a head office installation collecting files from various divisions of a corporation to build a data base). Both the development of formal and informal computer networks and the economic feasibility of large data bases are favoring the development of arrangements for a considerable volume of data interchange, whether directly over communication systems such as AUTODIN, or by the dispatch of reels of tape and boxes of cards. These are very significant areas of growth that are just beginning to emerge in commercial EDP and are already creating problems in data interchange within the federal government.

The development of data interchange is straightforward when the correspondents have agreed on the format. But where there

has been no prior agreement, conversion usually involves considerable manual intervention (Figure 8). Some typical problems are that:

(1) The sender's format may not be specified rigorously and an informal description may have to be debugged.

(2) The sender's format may not be expressible in the receiver's system.

(3) The sender's format descriptions may be embedded in the program.

(4) The format in the sender's system may vary from record to record and be embedded in the data.

Any of these problems may arise when either an existing application is converted to a new system, or a new member of a cooperating network has a system different from that of any existing member.

There are two basic problems:

(1) Few existing systems have any ability to deal with a new format automatically, and those that do are limited to data described in the same system.

(2) The number of different, and often incompatible, ways of describing data is increasing; e. g., format statements in FORTRAN; Data Description division in COBOL; COMPOOL in JOVIAL; File Format Table in the Formatted File System (FFS). Any solution to this problem should not restrict participants in the use, within their own local system, of any internal data structure or any programming or query language they like. It is expected that systems should interface with a limited set of known ways of describing data for interchange and provide conversion processes in their interfaces. If a suitable interface

- PROBLEMS BETWEEN EXISTING SYSTEMS
 - SENDERS FORMATS MAY NOT BE DEBUGGED
 - SENDERS FORMAT MAY NOT BE EXPRESSIBLE IN RECEIVERS SYSTEM
 - SENDERS FORMAT MAY BE EMBEDDED IN THE PROGRAM
- PROBLEMS BETWEEN NEW/EXISTING SYSTEMS
 - INABILITY OF AN EXISTING SYSTEM TO AUTOMATICALLY ADAPT TO NEW FORMAT
 - INCREASING DIVERSITY IN WAYS OF DESCRIBING DATA
 - e.g., JOVIAL: COMPOOL
 - FORTRAN: FORMAT STATEMENT
 - COBOL: DATA DIVISION
 - FFS-DMS: FILE FORMAT TABLE

FIGURE 8
PROBLEMS OF DATA INTERCHANGE

is to be developed, we will not want to standardize formats, which would be absurd, but we would want to standardize ways to describe formats. We also will want to attach the data descriptions to the data, so that the transmission of both data and its description can be performed without manual intervention. Figure 9 summarizes the requirements for a "data interchange or description language."

A data description language for data interchange does not principally have to be read and understood by humans. It can be thought of as a complicated coding to be generated and interpreted by the interface modules of systems in a network. In a well-designed system a user would describe data in the form provided for local use, and the system would translate to data interchange conventions. Therefore, the data description language should be generally compatible with data descriptions in current programming languages. Later, developments in programming languages may be influenced by a desire to remain compatible with data interchange conventions. In addition, a node in a network may occasionally be a human; any data description language should be reasonably understandable by humans.

It is not reasonable to have only one standard way to describe data for interchange. For example, there are at least two basic types of data structure in existence:

- (1) Hierarchically structured formats
- (2) List- and ring-structured formats.

- MUST BE:
- NON-RESTRICTIVE OF INTERNAL DATA STRUCTURES
 - GENERALLY COMPATIBLE WITH DATA DESCRIPTIONS IN CURRENT LANGUAGES
 - SOME CAPABILITY FOR HUMAN INTERPRETATION
- ⇒ PROVIDE FOR:
- STANDARDIZED WAYS TO DESCRIBE FORMATS
 - AUTOMATICALLY ATTACHING DATA DESCRIPTION TO DATA
- i.e.:
- A COMPLICATED CODING TO BE GENERATED AND INTERPRETED BY INTERFACE MODULES

FIGURE 9
DATA DESCRIPTION LANGUAGE REQUIREMENTS

Figure 10 suggests that consideration should be given to developing a standard for data interchange description that:

- (1) Defines a limited set of data description languages
- (2) Specifies the conventions for locating the description of data in file-labels and record-headers.

SUMMARY

The management information systems of the future need to have the following major characteristics which are summarized in Figure 11:

- (1) A basic internal data management facility of some kind that can be adapted and suited to the kind of machine used, the kind of performance needed, and the volumes of data that are to be expected. The internal strategy can be linked lists, inverted files, or others, as necessary.

- (2) A whole variety and family of languages for users. Different levels of language for different levels of sophistication of users and different kinds of languages for different kinds of users. There must also be some overall pattern, so they can all be translated to some basic language that operates directly on the internal data management systems. In addition, some compatibility among them must exist so that anyone who has to use several of them will not find great differences when he moves from one to another.

- (3) Conventions and techniques which we have characterized as a data description language in order to be able to transmit data between different systems and between different parts of systems.

● SELECT A LIMITED SET OF DATA DESCRIPTION LANGUAGES

MARC 11 (USASI Z-39)

COBOL EXTENSION

ACS

● SPECIFY CONVENTIONS FOR LOCATING THE DATA DESCRIPTION WITHIN
THE DATA

FIGURE 10
STANDARDIZATION - - FIRST STEP

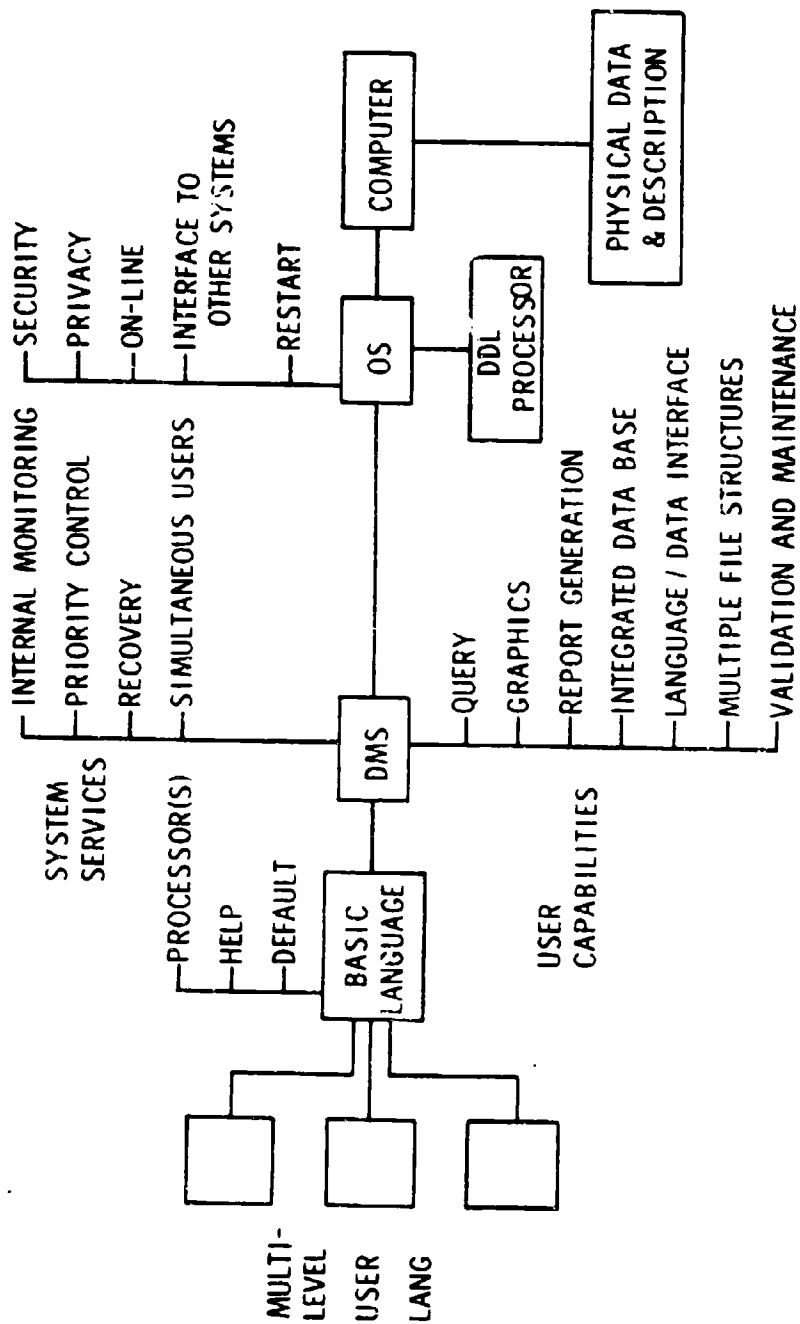


FIGURE 11
4TH GENERATION MIS CAPABILITIES

FURTHER READING

For those who would like to pursue these topics further, we append a list of references. References 1, 2, 3 and 4 are surveys for data management systems which give a good view of the kinds of data management systems available today.

A general discussion of the problems of software compatibility including those of data bases in networks and of families of languages of different users is presented in Reference 5, which also includes an extensive bibliography on the subject.

An example of typical on-line system interface is described in Reference 6.

A general survey of some simple on-line data management systems is given in Reference 7.

REFERENCES

1. J. Minker and J. Sable, "File Organization and Data Management," in Annual Review of Information Science and Technology, Volume 2, C. A. Cuadra, ed., New York, J. Wiley & Sons, 1967.
2. J. P. Fry, S. Bramson, D. C. Fried, W. P. Grabowsky, J. Jeffries, Jr., S. Mahle, H. J. Sternick, Data Management Systems Survey, The MITRE Corporation, MTP-329, January 1969, AD-684707.
3. CODASYL Systems Committee Technical Report, A Survey of Generalized Data Base Management Systems, ACM, May 1969.
4. J. Fry, J. A. Gosden, Survey of Management Information Systems and Their Languages, The MITRE Corporation, MTP-313, May 1968, AD-684706.
5. J. A. Gosden, "Software Compatibility: What Was Promised, What We Have, What We Need," Proc. Fall Joint Computer Conference, 1968, AFIPS Volume 33, Part I, The Thompson Book Co., Washington, D. C., December 1968.
6. H. P. Burnaugh, The BOLD (Bibliographic On-Line Display) System, Systems Development Corporation, Santa Monica, California, 6 April 1966, AD-632473.
7. N. Welch, A Survey of Five On-Line Retrieval Systems, The MITRE Corporation, MTP-322, August 1968, AD-686812.